韓國 컴퓨터情報學會 論文誌
第9卷 第2號, 2004. 6.

2004-9-2-1-8

멀티미디어 데이터베이스 환경에서 시각화된 사용자 정의 스키마 통합

이 현 창*

Visual User Defined Schema Integration at Multimedia Database Environment

Hyun Chang Lee*

요 약

최근 데이터베이스를 이용한 정보처리응용 시스템이 증가하고 있는 추세이다. 그러나 많은 데이터를 가지고 있지만 개별적이면서 관련성이 적은 수많은 데이터를 가지는 기업은 데이터를 종합적으로 분석 처리하여 사용자 요구에 신속하게 응답할 수 있도록 다양하고 고품질의 정보 서비스를 지원하기 위한 데이터를 필요로 한다. 이를 위해 데이터 웨어하우스 환경에 관한 연구가 많이 이루어지고 있다. 본 논문에서는 최종 사용자 요구에 대처하기 위해 비주얼 환경을 사용한 데이터 웨어하우스의 효과적인 설계 및 구현 결과를 보인다. 또한 기존의 일반 SQL 질의를 수행할 수 있도록 한다.

Abstract

In these days, application systems for processing information using database is increasing. Enterprises holding a lot of data do not possess needed data but instead include unrelated, independent and individual data. As a result, it only contains disparate data. Disparate data is ambiguous and it does not support current integrated information. In response to the above problems, data warehouse may provide a solution. Building a data warehouse needs a systematic design because of its complexity. This paper describes an efficient design methodology using visual environment for data warehouse to cope with the requirements of end users. Also, the system is able to process existential SQL query.

▸ Keyword : data warehouse, visual, schema, integration

이 논문은 2004년도 한세대학교 교내학술연구비 지원에 의하여 연구되었음.

* 한세대학교 컴퓨터공학과 조교수

Ⅰ. 서 론

오늘날 각 기업의 정책 결정자들은 그들이 필요로 하는 정보 요구에 대해서 실제로 여러 단계의 정보 가공을 거쳐야 하거나, 아니면 원하는 정보를 얻을 수 없는 상태이다. 또한 이들은 방대한 규모의 기업이 소규모 비지니스의 활동처럼 민첩해지기를 바라며, 기업 차원에서 소요되는 경비가 정상적인 소요를 필요로 하는지를 파악하여 투자대 회수(return on investment, ROI) 분석을 요구하게 되었고, 이러한 투자대 회수의 수행과정에서 위험, 비용, 시간, 장점 등의 문제가 제기되었으며, 특히 의사 결정 지원 시스템(decision support system : DSS)의 필요성이 증가함에 따라서 기존의 데이터베이스에서도 이와 같은 시스템을 더 좋고, 빠르고 효율적으로 지원하기 위한 요구사항이 대두되게[1][2][3] 되었다.

이와 같은 요구에 대응하여 데이터 웨어하우스라는 방법론이 대두되었으며, 웨어하우스라는 것은 위의 문제들에 대한 해답을 제공한다. 따라서 웨어하우스는 한 기관의 현재 상태 데이터, 뿐만 아니라 과거의 데이터도 유지하므로 누적된 통합 데이터를 유용한 정보로 변환하여 저장하는 대용량의 정보 저장고[2][4]라고 할 수 있다.

이러한 대용량의 정보를 포함하는 데이터 웨어하우스 구축을 위해서는 기존의 관계형 데이터베이스에서 개념적으로 표현된 데이터 모델은 기존의 운용 데이터 모델(Operational Data Model)로서 필요한 정보를 추출하여 의사 결정에 적합하도록 표현된 데이터 모델은 아니다. 그러므로 의사 결정을 위해서 데이터 웨어하우스에서는 차원 모델(Dimensional Model)[1][2][3]을 포함한다.

그러므로 본 논문에서는 차원 모델에서 사용자가 직접 선택 지정한 스키마를 중심으로 의사 결정을 원활하게 지원할 수 있도록 데이터 웨어하우스 환경에 적절한 스키마 생성기의 설계 및 구현 내용에 관하여 살펴본다. 또한 요즈음 인터넷 사이트에서 서비스 변화에 따른 본 연구의 필요성을 (그림 1)에 간략하게 도식화하였다.

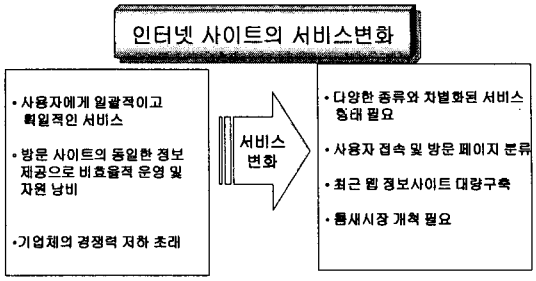

그림 1. 서비스 변화에 따른 연구의 필요성
Fig. 1 Necessity of research according to service transition

본 연구의 구성은 먼저, 데이터 웨어하우스 지원을 위해 전체적인 스키마 생성기 개발 구조를 살펴보며, 3장에서는 일반적으로 수행되는 스키마 생성 인터페이스에 관하여 기술한다. 4장에서는 본 논문의 핵심이 되는 의사 결정을 지원하기 위한 스키마 생성 인터페이스에 관하여 살펴보며, 5장에서 ODBC를 통한 데이터 웨어하우스 연결 인터페이스를 기술하며, 6장에서 결론 및 향후 연구 방향을 살펴본다.

Ⅱ. 사용자 정의 스키마 모듈

본 장에서는 웨어하우스를 위한 사용자 정의 스키마 생성기의 설계에 필요로 하는 각 모듈들을 제시하며, 이들에 대해서 살펴본다. 의사 결정을 지원하기 위해서 스키마 생성과 관련된 각 모듈을 (그림 2)에 나타내었다.

(그림 2)에서 일반적인 데이터 정의어(general DDL)를 통하여 기존 방법을 이용한 스키마 생성이 수행될 수 있으며, Dimension/Fact DDL을 이용한 스키마 설계는 의사 결정을 지원하기 위해서 데이터 웨어하우스 환경에 적절한 스키마 생성을 이룰 수 있는 경로이다. 이와 같이 스키마 설계는 의사결정을 효율적으로 지원하는 기능을 담당하는 모듈이다.

그러므로 스키마 설계는 각 주제 영역별로 주요 그룹과 보조 그룹 및 그룹간 연결자, 데이터 형태등이 논리적으로 형성되는 기능이 수행되어야 한다. 논리적인 스키마 설계가 이루어진 이후에 사용자는 GUI(Graphic User Interface)를 이용하여 보다 쉽게 스키마 생성을 수행할 수 있다.

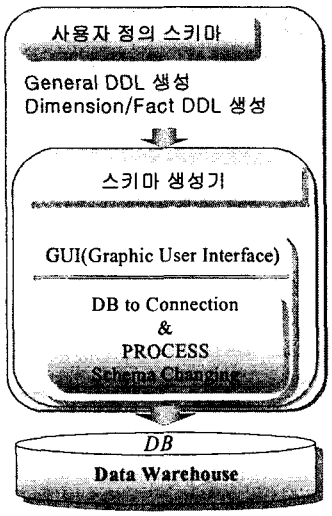

그림 2. 스키마 생성 개발 모듈
Fig. 2 Modules for developing schmea generator

그래픽 사용자 인터페이스(GUI) 모듈에서는 논리적으로 형성된 스키마 설계를 바탕으로 개념적인 스키마 생성을 이룰 수 있는 기능을 제공한다. 위에서 살펴본 것처럼 GUI 모듈에서도 일반적인 스키마 인터페이스 기능 제공, 다시 말해, 데이터베이스내에 원할하게 사용자가 정의한 테이블이 생성될 수 있도록 인터페이스 기능을 제공함과, 논리적으로 정의된 스타 스키마를 입력 받아 데이터 웨어하우스의 개념적 스키마를 생성하는 기능을 제공하여야 한다.

데이터 웨어하우스 스키마 생성기 모듈에서는 GUI를 이용하여 입력받은 정보를 바탕으로 데이터 웨어하우스 스키마를 생성한다. 다음의 각 장에서는 (그림 2)에서 보여주고 있는 데이터 웨어하우스를 위한 사용자 정의 스키마 생성기의 초기 화면을 (그림 2)의 모듈들 기반으로 수행이 이루어지게 된다. 먼저, (그림 2)의 전체적인 스키마 생성기 개발 구조에서 보여주는 흐름에 따라 각 모듈별로 살펴본다.

III. 일반 스키마 생성 인터페이스

본 장에서는 데이터베이스의 테이블 생성의 일반적인 수행에 관해 살펴본다. 먼저, (그림 3)에서는 기존의 데이터베이스 정의어(Database Definition language, DDL)를 이용하여 데이터베이스 "pubs"라는 곳에 새롭게 "sales"라는 테이블을 생성하기 위한 화면을 도시하고 있다.

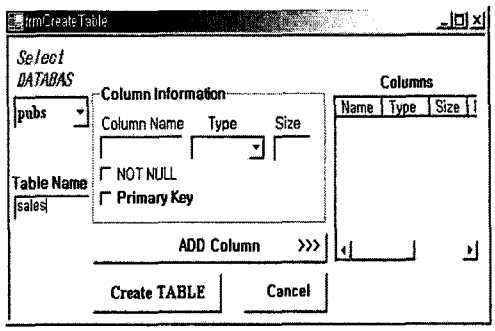

그림 3. "sales" 테이블 생성 화면
Fig. 3 Window for generating "sales" table

이와 같이 애트리뷰트 선언이 이루어진 후 "Create TABLE" 아이콘을 클릭하게 되면 커맨드라인에서 선언한 것과 같이 Informix 혹은 Oracle에서 사용하는 SQL 문장을 하나의 윈도우를 통하여 나타한 결과를 얻을 수 있다. 그림 4에서는 그림 3의 인터페이스를 통하여 생성 가능한 애트리뷰트 리스트를 보여주고 있다.

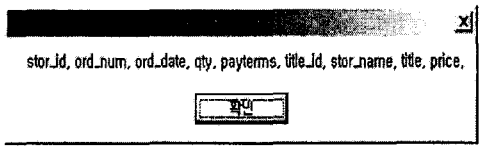

그림 4. "sales"테이블 애트리뷰트 리스트
Fig. 4 Attribute list for "sales" table

IV. 사용자 정의 스키마 생성 및 비교

본 장에서는 (그림 2)의 사용자 정의 스키마 생성을 위해서 Dimension/Fact DDL의 스키마 생성에 관하여 살펴보며, 기존과 간략히 비교 분석해 본다.

1. 분석용 테이블 생성 모델

개체-관계성(entity-relationship) 다이어그램들과 정규화 테크닉들은 일반적으로 OLTP 환경에서 데이터베이스를 설계하기 위해 사용되었다. 그러나 이들 ER 다이어그램에

의해 나타낸 데이터베이스 설계는 질의할 때나 데이터를 로딩(loading)할 때 효율성을 중요시하는 의사 지원 시스템을 구성하기에 부적합하다. 특히, 트랜잭션 처리 데이터베이스에서 사용된 정규화된 구조는 빠른 처리 결과와 분석에 맞게 설계되지 않았다. 반대로 의사 결정 지원 데이터베이스에서는 동향 및 시간 지원을 포함하기 때문에 상대적으로 크기가 크고 분석에 따른 시간이 많이 소요된다.

다음 (그림 5)와 (그림 6)에서는 의사 결정 지원 시스템을 구축하기 위해 메뉴에서 MSSQL 서버에 연결하기 전과 후에 나타난 화면이다.

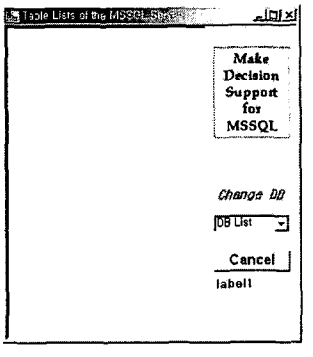

그림 5. DB 선택전 화면
Fig. 5 Window before selecting DB

상기 (그림 5)에서 "Change DB"의 데이터베이스를 변경하면 기존 수행 방법과 다르게 여러 데이터베이스의 내용을 선택할 수 있는 장점이 있으며, 이를 사용자가 선택할 수 있다.

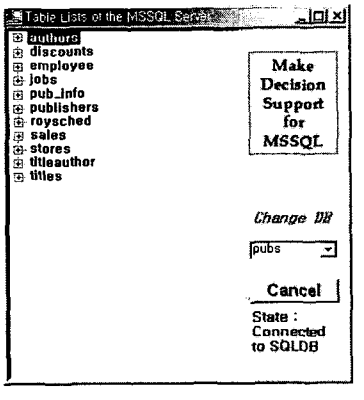

그림 6. MSSQL 데이터베이스 선택 전과후
Fig. 6 Window after selecting DB

(그림 5)와 (그림 6)에서는 데이터베이스를 선택하였을 때 테이블 리스트를 보여주고 있다. 그림에서 의사 결정을 위한 아이콘을 클릭하게 되면 다음 (그림 7)이 나타나게 된다.

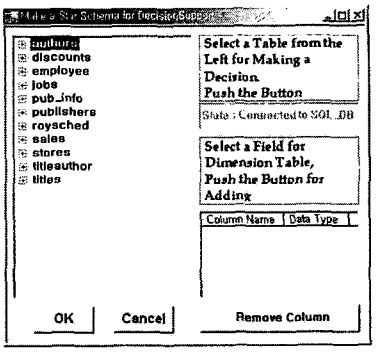

그림 7. 의사 결정을 위한 테이블 선택
Fig. 7 Table selection window for decision making

상기 (그림 7)에서는 의사 결정을 수행하기 위해서 먼저 중심 테이블을 선택하기 전 화면을 도시하고 있다. 이때 사실 테이블을 선택하고 사실 테이블 "sales"의 애트리뷰트 리스트가 다음 그림과 같이 나타난다.

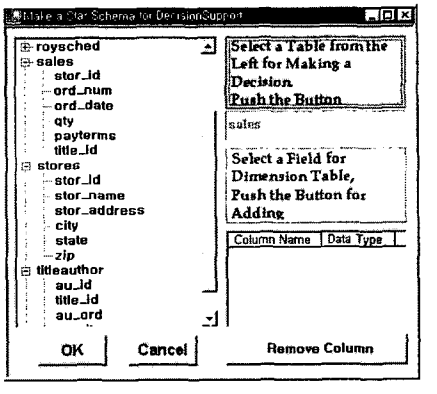

그림 8. 사실 테이블과 애트리뷰트 리스트
Fig. 8 Fact table and attribute list

(그림 8)에서 알 수 있듯이 테이블 "sales"에 나타난 애트리뷰트 리스트는 트랜잭션 처리에 맞게 존재하는 항목들이다. 이 테이블을 사실 테이블로 선택하고 난 이후에 분석을 위해 차원 테이블들의 필드들을 선택하도록 한다. 다음 절에서는 분석을 위한 사용자 정의 애트리뷰트 선택에 관하여 살펴본다.

2. 분석용 애트리뷰트 사용자 정의 모델

본 절에서는 사용자로부터 분석에 필요한 애트리뷰트를 다른 테이블들로부터 선택하고, 선택된 애트리뷰트를 (그림

8)에서 선택한 테이블에 추가함으로서 분석을 수행할 수 있도록 한다.

(그림 9)에서는 차원 테이블들의 애트리뷰트 리스트에서 의사 분석을 수행하기 위해 차원 테이블에 존재하는 애트리뷰트들을 사실테이블인 "sales"에 추가한 화면을 도시하고 있다. 그림에서 볼 수 있듯이 추가된 애트리뷰트들 중에서 "stor_name"은 차원 테이블 "stores"와 "city"는 상점 이름과 상점이 위치한 시의 이름 애트리뷰트의 속성을 가지는 것이며, 해당 속성을 우측하단에 나타내고 있다.

이와 같은 방법을 통해서 사용자는 자신이 원하는 분석을 수행하기 위해서 직접 스키마 선택 및 정의를 통하여 의사 결정을 수행할 수 있는 시스템 환경을 구축할 수 있게 되며, 기존의 일률적으로 구축된 방법과 다양한 결과를 얻을 수 있게 된다.

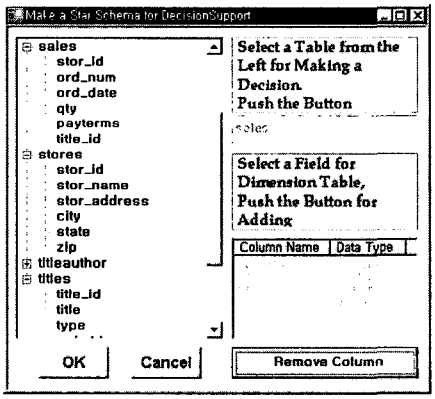

그림 9. 차원 테이블에서 애트리뷰트 선택
Fig. 9 Selected attributes in dimension table

다음 (그림 10)에서는 사용자가 정의한 애트리뷰트들이 추가된 "sales" 테이블을 나타낸 결과를 도시하고 있다.

그림 10에서는 사용자가 선택한 애트리뷰트 리스트가 분석을 위해 선택한 테이블에 추가된 스키마 결과를 보여주고 있으며, 사용자는 이 테이블을 통해 원하는 데이터 분석을 수행할 수 있는 환경을 마련할 수 있게 된다.

이와 같이 분석용 테이블 스키마는 사실 테이블(fact table)과 차원 테이블(dimension tables)의 두가지 형태의 테이블을 포함하고 있다. 사실 테이블은 기업에 관한 정량적이면서 사실적인 데이터를 포함한다. 이 정보는 주로 수치 측정치로 구성된다. 차원 테이블은 사실 테이블에 비해서 작으며, 기업의 차원에 영향을 미치는 기술(descriptive) 데이터를 가지고 있다. SQL 질의는 사실 테이블과 차원 테이블 사이에 미리 정의된(predefined)것이나 사용자가 정의한 조인 패스를 사용하여 데이터상에 제약을 가하여 선택된 정보를 반환한다.

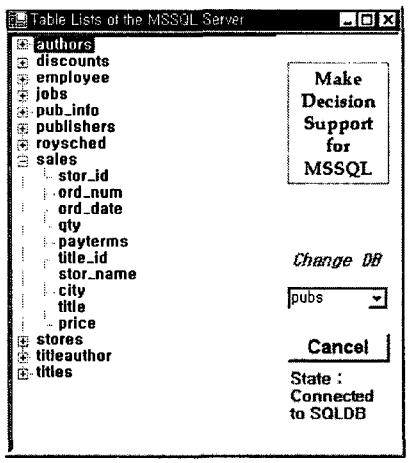

그림 10. 분석용 애트리뷰트가 추가된 스키마
Fig. 10 Schema added attributes for analysis

V. 결론 및 향후 연구 방향

데이터 웨어하우징의 장점중 하나는 기존의 스키마를 변경하지 않고도 새로운 스키마인 사실테이블과 차원테이블을 삽입할 수 있다는 것이다. 이러한 환경 하에서 사용자의 요구에 적절히 대처할 수 있기 때문에 본 연구에서는 의사 결정을 지원하기 위하여 사용자가 정의한 스키마 생성기를 설계 및 구현하였다.

본 연구의 시스템에서는 다양한 데이터베이스 환경을 지원하기 위해서 MSSQL과 많은 일반 사용자가 사용하는 Access 관계형 데이터베이스를 기반으로 구현하였으며, 본 연구에서 보여준 내용은 MSSQL 데이터베이스를 기준으로 결과를 살펴보았다. 또한, 하나의 사실 테이블을 중심으로 분석용 애트리뷰트를 위해서 하나 이상의 테이블을 선택할 수 있으므로 사용자는 의사 결정에 요구되어지는 관련 테이블들을 테이블 리스트에 등록함으로써 의사결정을 보다 신속하게 지원할 수 있는 스키마를 생성할 수 있게 된다. 본 시스템을 바탕으로 요약(summary)과 집단화(aggregate) 기능을 제공하게 되면 더욱 빠르며, 정확한 사용자 질의 결과를 얻을 수 있을 것이다.

참고문헌

[1] W.H. Inmon and R.D. Hackathorn, Using the Data Warehouse, John Wiley & Sons, 1994.
[2] R. Kimball, Data Warehouse Toolkit, John Wiley & sons, 1996.
[3] Red Brick System, "Star Schemas and STAR join Technology", Red Brick Systems White Paper, 1996.
[4] Informix System, "Designing the Data Warehouse on Relational Database", Informix Systems White Paper, 1996
[5] W.J.Labio, Y.Zhuge, J.L.Wiener, H.Gupta, H.Garcia-Molina, J.Widom. The WHIPS Prototye for Data Warehouse Creation and Maintenance*. In SIGMOD Conference p.p.557-559, 1997.

저 자 소 개

이 현 창

1993년 원광대학교 컴퓨터공학 (학사)
1996년 홍익대학교 전자계산학과 (석사)
2001년 홍익대학교 전자계산학과 (박사)
2001년 ~ 2003년 8월 경인여자대학 조교수
2003년 9월 ~ 현 재 한세대학교 컴퓨터공학과 조교수
〈관심분야〉 웹 정보 시스템, 데이터 웨어하우징, 주기억장치 시스템, 모바일 응용